User Manual

For
openCONFIGURATOR

Prepared By
Kalycito Infotech Pvt Ltd.,
India

Identifier	User_Manual_openCONF	**Version**	1.04
Prepared By	Ramakrishnan P	**Date**	05-Jul-2013
Approved By	Vinod PA	**Confidentiality**	Public Domain Document

Document Revision History

Version	Date	Modified By	Remarks
0.01	31-Mar-2009	Powerlink Team – Kalycito	Initial Draft
0.02	09-Apr-2009	Powerlink Team – Kalycito	Incorporates information for Vista
1.00	16-Apr-2009	Powerlink Team – Kalycito	Incorporates Known bugs and their work around and compilation - how to
1.01	08-May-2009	Powerlink Team – Kalycito	Updated known bugs and their updated work around(s). New Notes to the user are added.
1.02	22-May-2009	Powerlink Team – Kalycito	Added installation procedure for RPM and Debian based systems and few cosmetic changes
1.03	25-Apr-2013	Ramakrishnan P	Major changes to the document architecture, Updated with the features & bug fixes till v1.3rc1
1.04	05-Jul-2013	Ramakrishnan P	Updated with the features & bug fixes for the v1.3rc2. Added several new sections.

License

Registered Office:

Kalycito Infotech Private Limited,

E-LAB, Science & Technology Entrepreneurial Park I,
PSG College of Technology, Avinashi Road,
Coimbatore – 641004, Tamil Nadu, INDIA

Phone 00 91 422 4518454
Email: info@kalycito.com

Sales Office:

Kalycito Infotech Private Limited,

479-B1, Avinashi road, Peelamedu,
Coimbatore – 641004, Tamil Nadu, INDIA
Phone 00 91 422 2565644
Email: sales@kalycito.com

Table of Contents

Figures

Table Index

Abbreviations

API	Application Process Interface
CAN	Controller Area Network
CDC	Concise Device Configuration
CiA	CAN in Automation
CN	POWERLINK Controlled Node (slave)
DLL	Dynamic Link Library
EPL	Ethernet POWERLINK
EPSG	Ethernet POWERLINK Standardization Group
GUI	Graphical User Interface
ID	Identifier
IEC	International Electro technical Commission
MN	POWERLINK Managing node
MNOBD	Managing node's Object Dictionary
NMT	Network Management
PDO	Process Data Objects
Preq	Poll Request (POWERLINK frame type)
Pres	Poll Response (POWERLINK frame type)
RPDO	Receive Process Data Object
SWIG	Simplified Wrapper and Interface Generator
TCL	Tool Command Language
TPDO	Transmit Process Data Object
XAP	Extend Application Process variables
XDC	XML Device Configuration file
XDD	XML Device Description file
XML	Extensible Markup Language

1. Introduction

1.1. Purpose

This document is intended for the users of openCONFIGURATOR-V-1.3.0 tool.

1.2. Intended audience and reading suggestions

User is assumed to possess basic knowledge on openPOWERLINK.

1.3. Operating environment

This tool is designed for the following operating environments:

- Various Linux distributions
- Windows XP
- Windows Vista
- Windows 7

2. Key Features

- Generates the CDC for the openPOWERLINK network
- Generates the network variables in
  - xml [xap.xml]
  - header file [xap.h]
  - C# .Net struct [ProcessImage.cs]
- The Process Image variables of the MN are as per the “CiA 302-4 CANopen additional application layer functions – Part4: Network variables and process image” specification
- Computes the MN PDO Mapping automatically
- Support Cross Traffic communication between the CNs
- Main package is available under BSD license

3. Setup - openCONFIGURATOR

Download latest version of openCONFIGURATOR from http://sourceforge.net/projects/openconf/

3.1. Linux

- **Install**
  - Please download the appropriate installer (32bit or 64 bit)
  - Un-tar the openCONFIGURATOR.tar.gz file
  - Open the terminal, and move to the extracted directory
  - To check & install the required packages, run
    sudo ./configure
  - If configuration succeeds, Makefile will be created
  - To install openCONFIGURATOR, run the following command from the terminal
    sudo make install
- **Launch**
  - From command prompt:
    - Open the terminal
    - To launch, type *openCONFIGURATOR*
  - From GUI:
    - Go to Applications > Programming
    - Click on '*open*CONFIGURATOR'

3.2. Windows

- **Install**
  - For Windows (XP, Vista & 7), Please install the ActiveTCL version 8.5. The executable can be obtained from http://www.activestate.com/activetcl/downloads
  - Unzip the openCONFIGURATOR.zip file and double click on the openCONFIGURATOR_Setup.exe file and follow the instructions

  Note: For window7 & vista the setup should be run as Administrator [right click on the setup file and click on 'Run as Administrator']

Now the Installer Dialog will open as shown below

- Read through the License and if you agree, press **'I Agree'** button and proceed with the installation

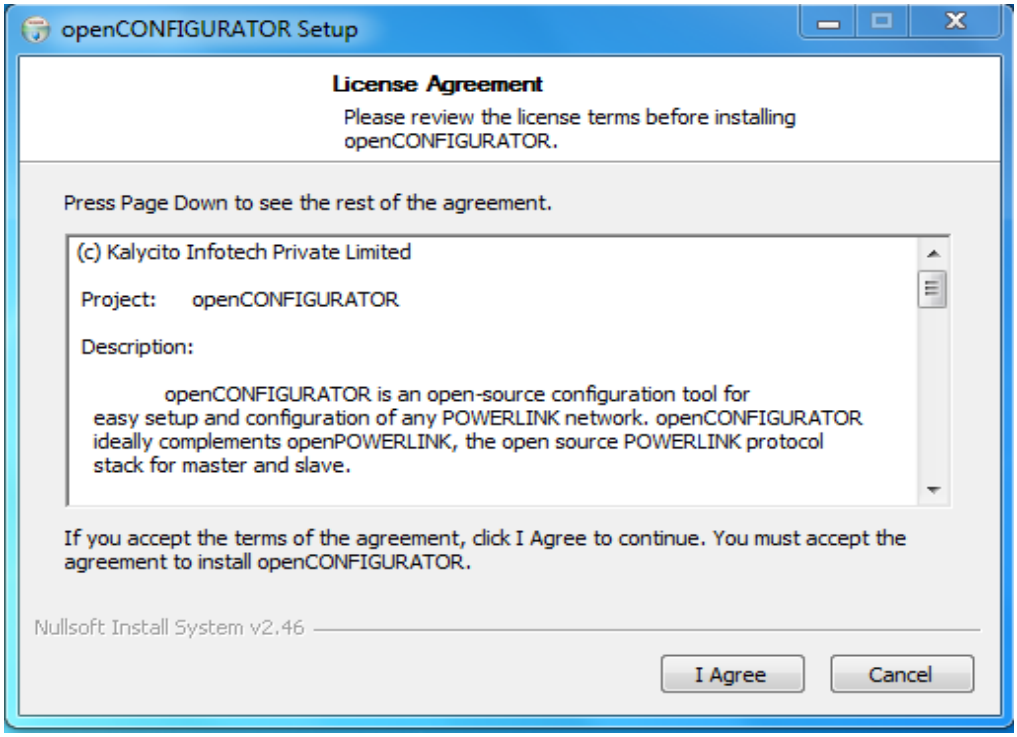

Figure 1: Installer - License page

- Press **NEXT** button to continue with the installation

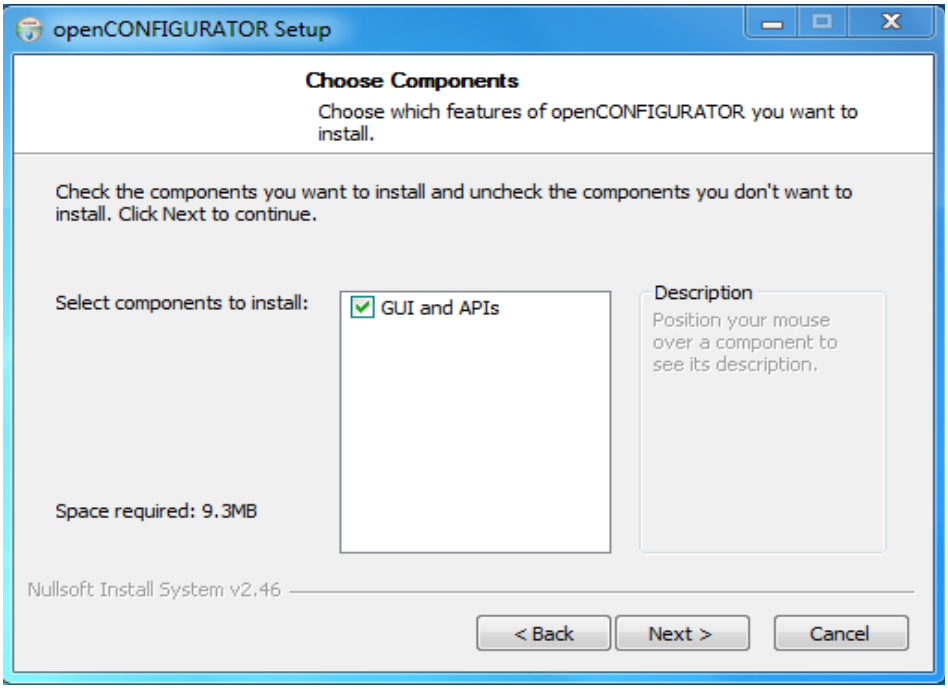

Figure 2: Installer - Components Page

- Select the directory where the tool should be installed. Click ‘Next’

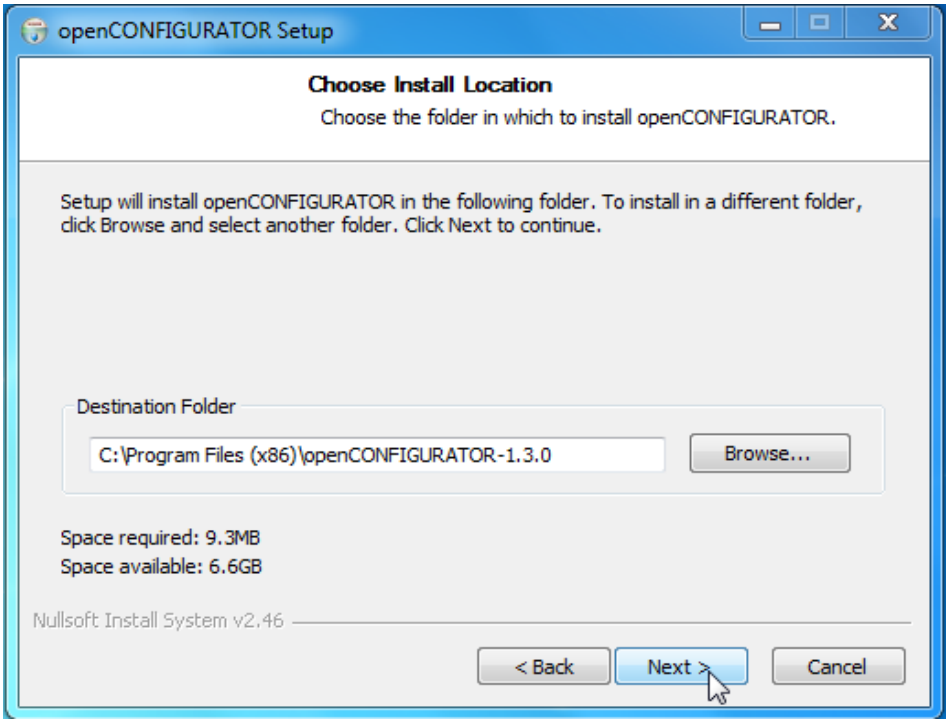

Figure 3: Installer - Install Path

- Tick “Do not create shortcuts” check box if you wish not to create start menu entry and Press **Install**

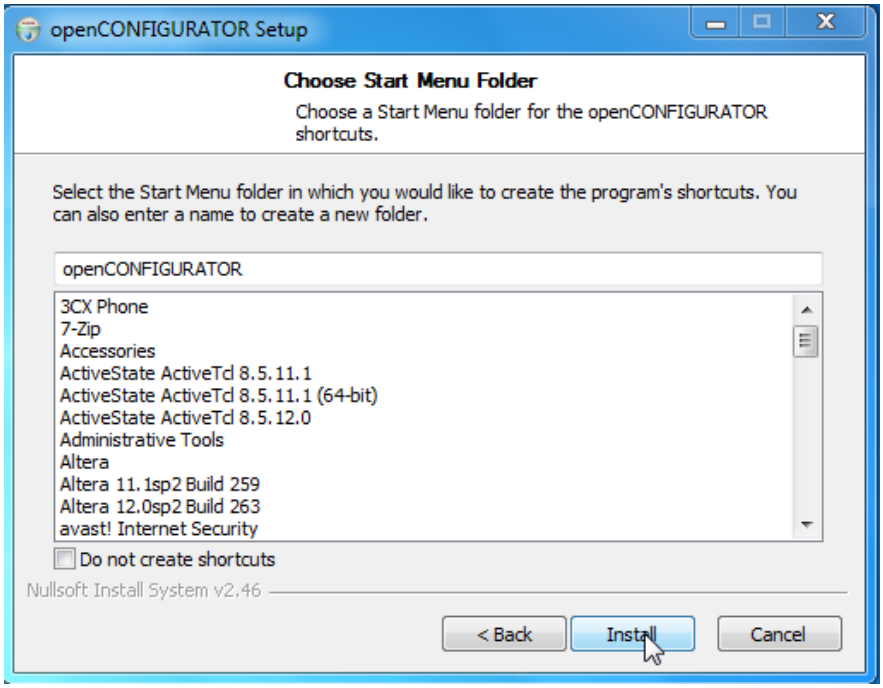

Figure 4: Installer - Start Menu

- Now the Installer will show a message than it installation is completed successfully

- **Launch**
  - Go to Start Menu > All Programs > openCONFIGURATOR
  - Click on openCONFIGURATOR

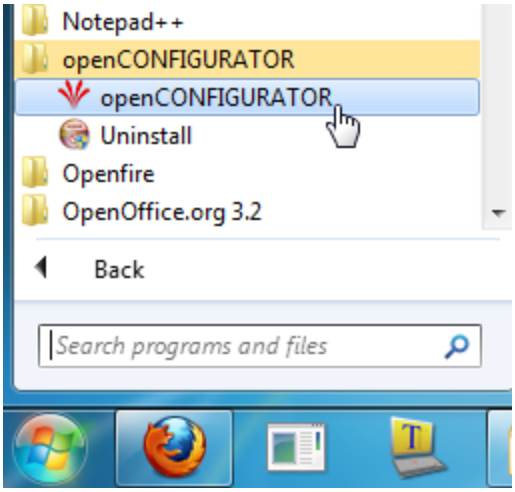

Figure 5: Windows - Launch Tool

Note: If you have chosen not to create shortcuts, you will not find the start menu entry. So you can launch the tool by double clicking on the openCONFIGURATOR.exe in the installation directory that you've set during the installation.

4. Using openCONFIGURATOR

4.1. File Menu

4.1.1. New Project

The user can create a new project by selecting **'File > New Project'** or by using the keyboard shortcut **'CTRL + N'**

4.1.2. Open Project

The user can open the already created projects by selecting **'File > Open Project'** or by using the keyboard shortcut **'CTRL + O'**

Note: It is highly recommended to use the projects created by newer versions.

4.1.3. Save Project

The project can be saved by selecting '**File > Save Project'** or by using the keyboard shortcut '**CTRL + S**' or by clicking on the 'Save' icon as shown in the below figure

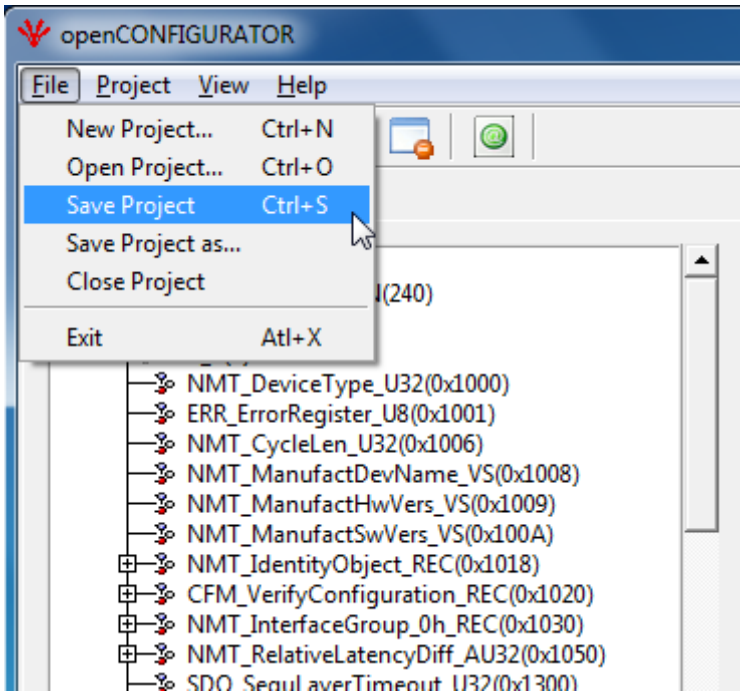

Figure 6: Save Project Menu

Save project will save the following files in the <project directory> */octx* folder

- One octx file for each CN present in the project. The name of the file is the NodeId of the CN
- One octx file for MN. Name of the file is 240(NodeId of MN)

Also on oct file will be saved for the project in the <project directory>.

4.1.4. Save Project As

The user can open save a copy of the projects by selecting **"File > Save Project As".** The tool will switch to the newly created copy.

4.1.5. Close Project

By selecting this option the tool will close the project and can be done by selecting **"File > Close Project**" as shown in the below figure

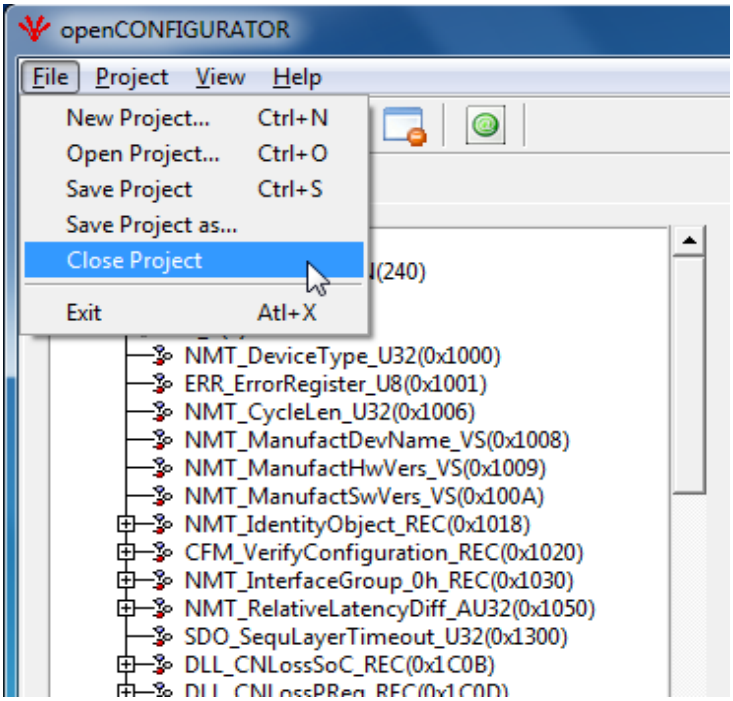

Figure 7: Close Project

4.2. Project Menu

4.2.1. Build Project

User can build the project by selecting the "**Project > Build Project**" or by using the function key "F7" as shown in the below figure

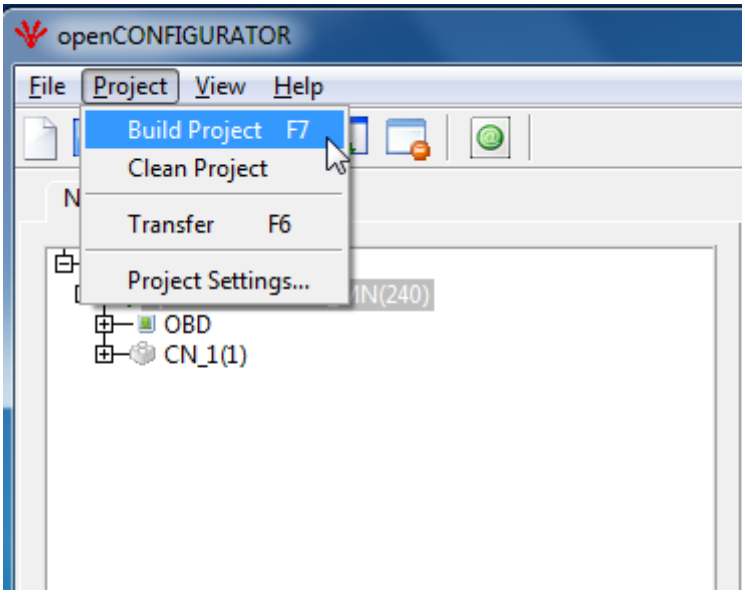

Figure 8: Build Project Menu

User can also build the project by clicking 'Build Project' icon as shown below

4.2.2. Clean Project

User can remove the output files (eg: mnobd.cdc, xap.h) from the project by selecting the "**Project > Clean Project**"

4.2.3. Transfer

The user can copy the output files generated by the tool to a fixed powerlink project directory directly from the tool in a single click.

- To achieve the copy operation, follow the below steps:
  - Set the destination path in the Transfet.bat(windows)/Trasfer.sh(Linux) file present in the installation directory
  - Now the user can copy the output files (eg: mnobd.cdc, xap.h) from the project by selecting the "**Project > Transfer"** or by using the Function key "F6"

4.2.4. Project Settings

The user can any time change the project settings of the tool by selecting the "Project->Project Settings" option.

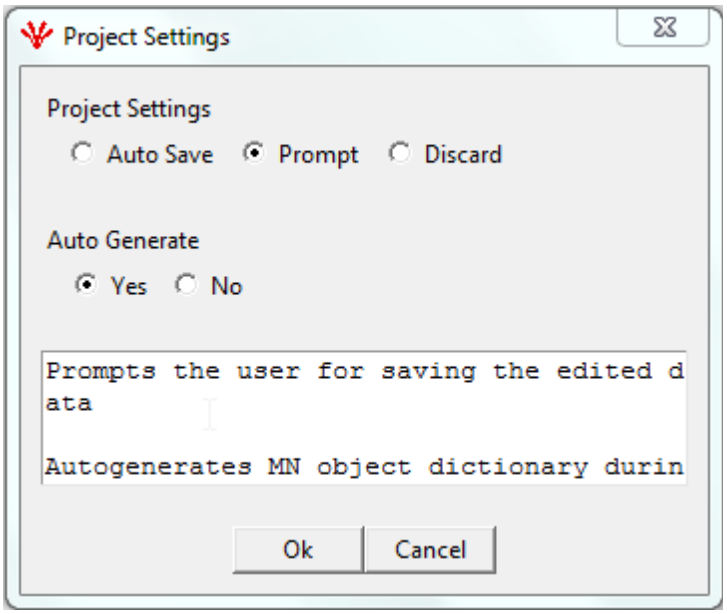

Figure 9: Project Settings Window

4.3. View Menu

The user can switch between the ‘Simple View’ & ‘Advanced View’ of the tree browser.

In ‘Simple View’, the below are visible in the tree browser,

- CN Name
- Node ID

In ‘Advanced View’, the below can be viewed by expanding the corresponding entries in tree browser

- Index of Node
- SubIndex of an Index
- PDO of a Node

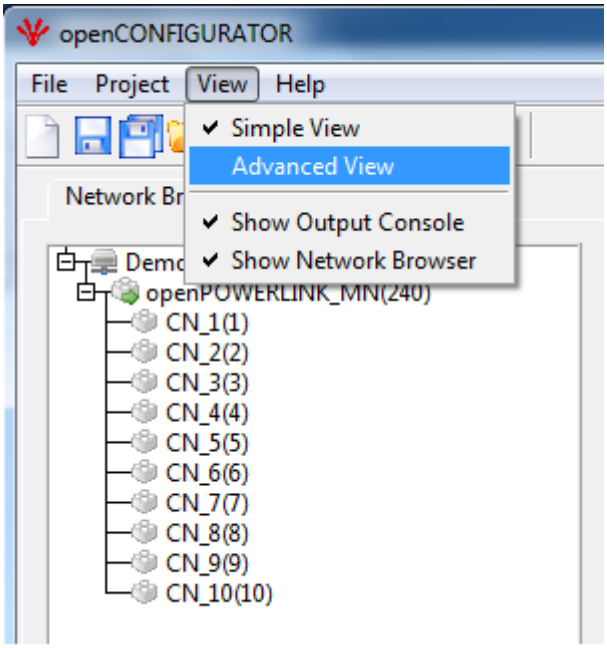

Figure 10: View menu

4.4. Console window

The user can view the status messages, warnings and error messages during build in the console window as shown in below

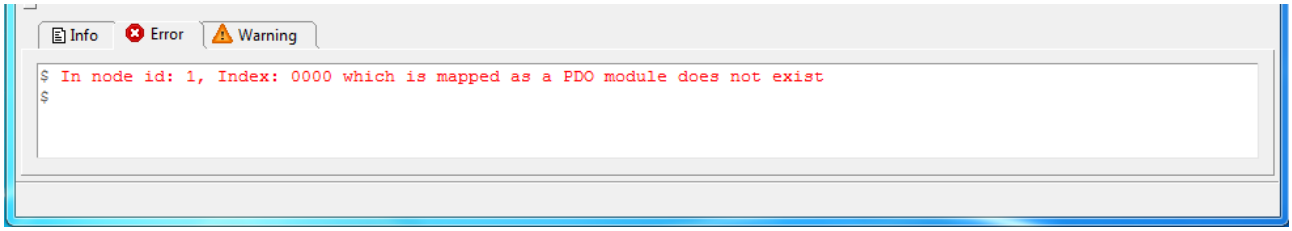

Figure 11: Console window

4.5. Project wizard

The project wizard helps you in creating a New Project or to open an Existing Project.

When the user launches openCONFIGURATOR tool it will ask either to 'Create New Project' or 'Open Existing Project' as shown in the below figure

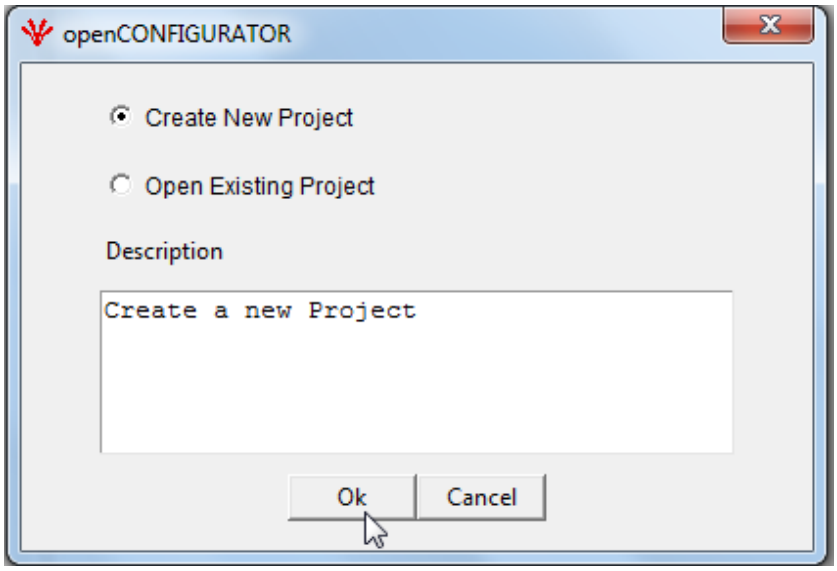

Figure 12: Create New Project

Also the user can create a new project from the menu bar by selecting ***File*** > ***New Project*** *or* ***File > Open Project***

4.5.1. Project Wizard - Name

The below options are to be entered/selected as applicable in the 'Project Wizard' dialog box

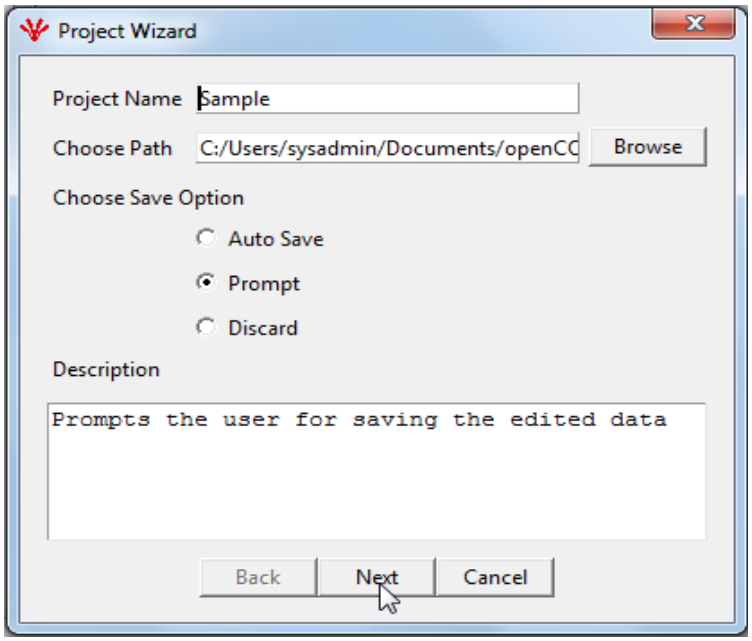

Figure 13: Project Wizard - Name

- **Project Name**
  - The maximum limit of the 'Project Name' is 32 characters and special characters & space are not allowed in the 'Project Name'

- **Choose Path**
  - User can select the location for placing the projects by clicking the 'Browse' button next to the 'Choose Path' field

  Note: The default path is the user's home directory/openCONFIGURATOR_Projects

- **Choose Save Option**
  - User can select the Project's save option as any one of the below

Save option	Description
AutoSave	Saves the configuration automatically without prompting the user
Prompt	Prompt to an user, to ask if user wants to save the data before exiting from the project
Discard	Discards any modifications made to the configuration. Manually the user can save the configuration by clicking save button

Table 1: Save options

Note: The user can change the "Save Type" at any time by clicking on the **Project > Project Settings**

4.5.2. Project Wizard – MN XDD

- After clicking on the “Next” button the Project Wizard – MNN XDD will appear as shown below.

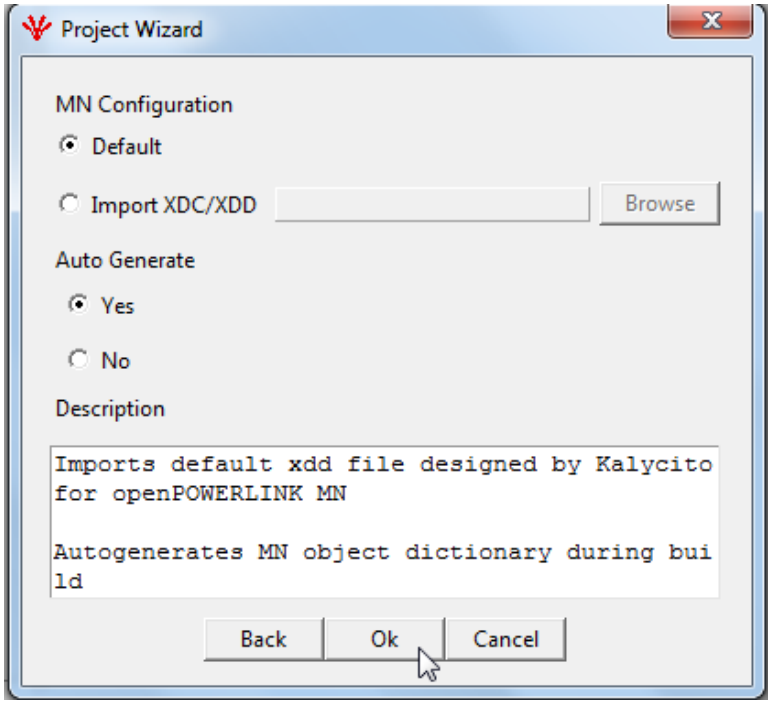

Figure 14: Project Wizard – MN XDD

- **MN Configuration**
  - There are two options by which MN OBD can be created, 'Default' (or) 'Import XDC/XDD'

Configuration Option	Description
Default	Default MN xdd which will be available with the installation package
Import XDD/XDC	User defined MN configuration

Table 2: MN Configuration

- **Auto Generate**

Auto Generate Option	Description
Yes	The MN configuration will be auto generated with the available CN's configuration
No	The MN configuration will have to be manually generated/updated by the user

Table 3: Auto Generate Option

  - **Auto Generate – Yes**
    - If 'Auto Generate' is set to 'Yes', the below will be automatically generated
      - PDO mapping for the Managing Node
      - The mapping configured in the CN will be mapped to the MN and the Process Image Variables will be created
        - Pres and Preq Payload length values
        - Offset for the CN if the station is chained.

Note:
An Object/Index is an array of SubObjects/SubIndices where each SubIndex corresponds to the CN with the Node ID equal to the SubIndex

- An alternate way to achieve “Auto Generate” mode to right click on the MN and select 'Auto Generate' as shown in the below figure

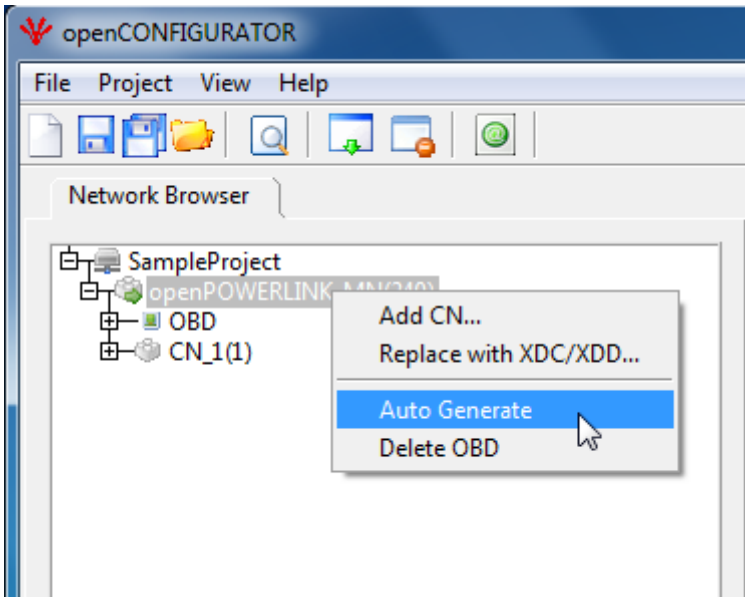

Figure 15: Auto Generate MNOBD

  - **Auto Generate – No**
    - If 'Auto Generate' is set to 'No', PDO mapping of MN will not be generated automatically and the user has to take care of the mapping and the changes in the MNOBD.
    - The changes made to the project will reflect in the CDC regardless of the correctness of the configuration data.

Note:
The user can change the ‘Auto Generate Mode’ at any time from the ‘Project Settings’ window by clicking on the **Project > Project Settings.** Refer Figure 9: Project Settings Window

4.6. Adding a CN Node

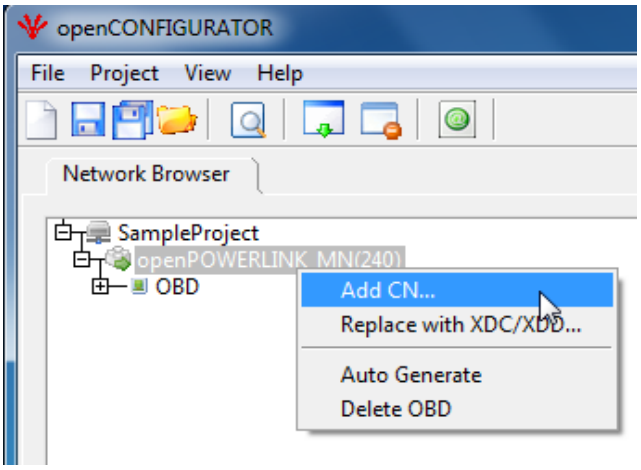

Figure 16: Add CN Menu

- A CN Node can be added by right clicking on the MN Node and selecting 'Add CN' option from the sub menu that appears as shown above.

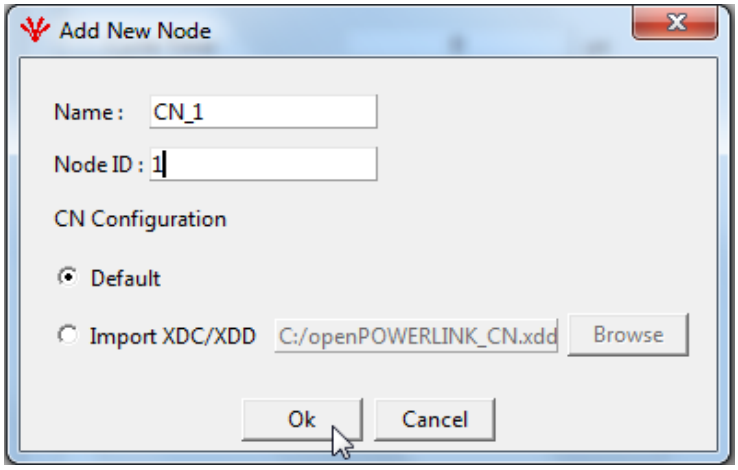

Figure 17: Add CN Window

- After clicking on 'Add CN' a pop-up will appear where user can enter CN configuration by referring the below table and can select the xdd/xdc files for that CN as shown above

New CN Configuration		Description	Range
Name		Name for the Node	1-32 Chars
Node ID (decimal value)		Node Id for the Node. Range(1 - 239)	1-239
CN Config.	Default	Default CN xdd which will be available with the installation package	
	Import XDD / XDC	User defined configuration for the CN. **Note:** Please validate your XDD with the XDD-Check tool (a free utility available in the link http://www.ethernet-powerlink.org)	

Table 4: New CN configuration

4.7. Adding an Index

- For an MN or CN, the Index can be added by right clicking the node in which Index has to be added as shown in the below figure

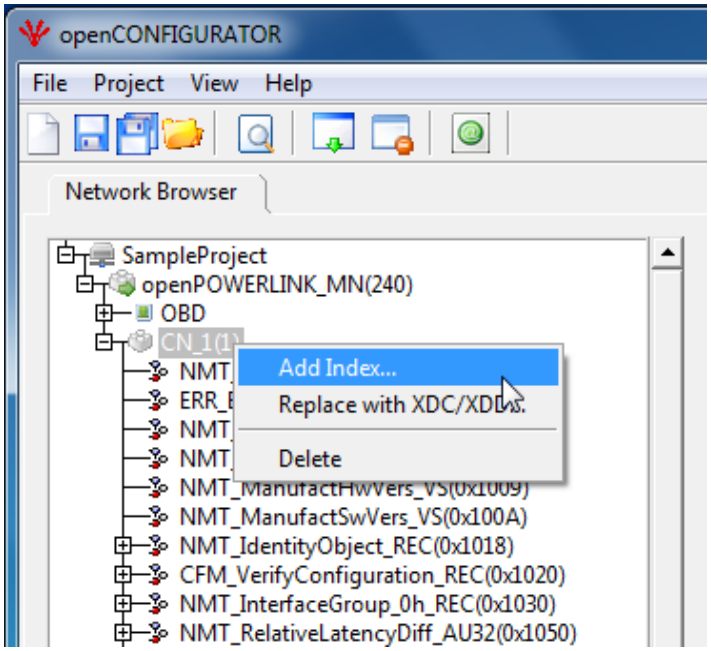

Figure 18: Add Index Menu

- After selecting the 'Add Index' menu a pop-up with a text box will appear as shown. The user can add an index by referring the table below

Name	Range (hex)	Applies to	Adds with default properties	Properties Editable in	
				MN	CN
Communication Profile Area Objects	1000 - 1FFF	MN & CN	Yes	Partial	Partial
Manufacture Specific Profile Area Objects	2000 - 5FFF	CN	No	Yes	Yes
Standardized Device Profile Area Objects	6000 - 9FFF	CN	No	Yes	Yes
Standardized Interface Profile Area Objects	A000 - BFFF	MN	No	No	Yes
Reserved for further use	C000 - FFFF	Reserved	No	Yes	Yes

Table 5: Index configuration

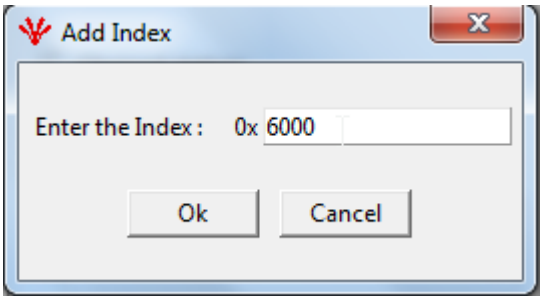

Figure 19: Add Index Window

- The user can enter the Index Id and press 'Ok'. The Index will be added in the Node as shown in the below figure

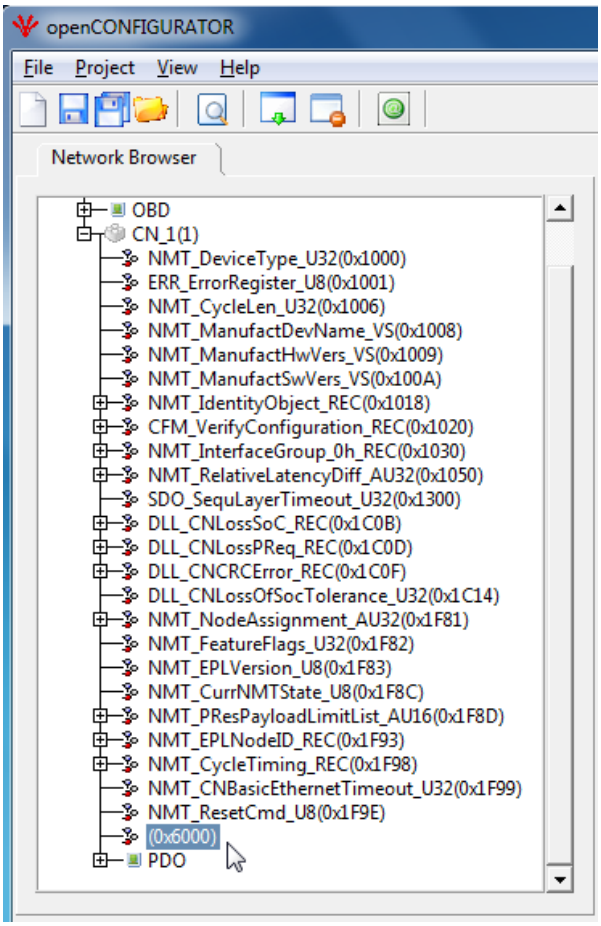

Figure 20: Index Added - Tree

- See "**Editing Object / SubObject Properties"** on how to add properties for the Index

Note:
Index Id's (0x14xx, 0x16xx, 0x18xx, 0x1Axx) will be added under the PDO node. The user can also add those indexes by right clicking on the PDO node.

4.8. Adding a SubIndex

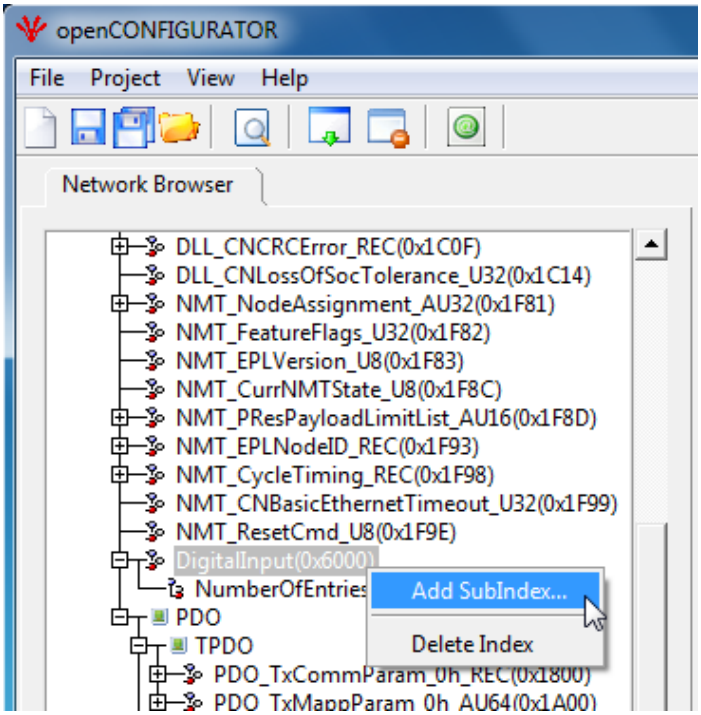

Figure 21: Add subindex Menu

- SubIndex can be added by right clicking the Index then select “Add SubIndex” as shown in the below & also refer to the Table 6: SubIndex configuration for more details

Name	Index Range	SubIndex Range	Default Properties	Properties Editable in	
				MN	CN
Communication Profile Area Objects	0x1000 - 0x1FFF	0x00 - 0xFE*	Yes	Partial	Partial
Other Objects	0x2000 - 0xFFFF	0x00 - 0xFE	No	Yes	Yes

Table 6: SubIndex configuration

** Refer EPSG_Specification*

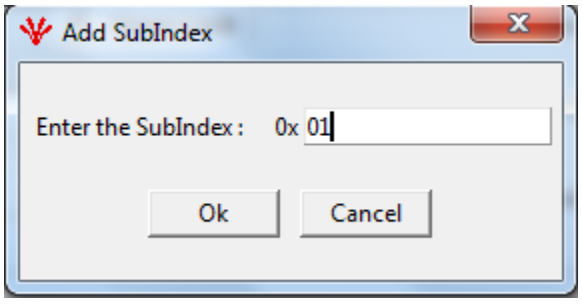

Figure 22: Add subindex

- Enter the SubIndex Id and press ‘OK’ SubIndex will be added under the parent Index

Note: The Objects with ObjectType Var and DefType cannot have subindexes

4.9. Editing Object / SubObject Properties

The user can edit the properties for the Objects / SubObjects by referring to the below table

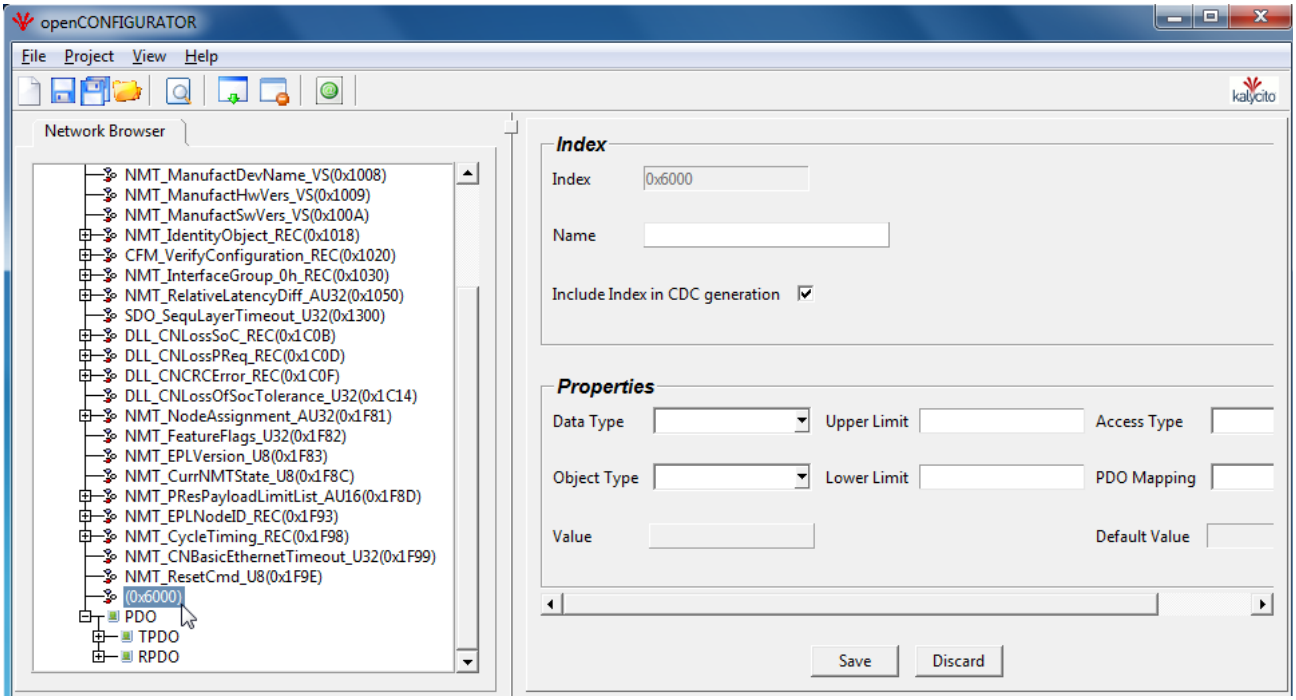

Figure 23: Edit an Object

- Index
  - The Index is the Id for the Object/Subobject which will be added while adding the Index/SubIndex. This cannot be changed after creating a Index/SubIndex.
  - Index shall be declared as hexadecimal value
  - Eg: 0x1F81, 0x25F4, 0x6201, 0xA480 – for Objects and 0x00, 0xFE – for SubObjects

- Name
  - Name provides a textual description of the function of that particular object
  - Name shall be in accordance to IEC 61121-3 standards
  - Total length of the name shall be equal or below 32 characters
  - Eg: NMT_FeatureFlags_U32, DigitalInput_U8, etc.,

- Include In CDC generation
  - Include in CDC check determines the inclusion of the actual value of the index/subIndex in the CDC.

Note:

To include a Sub-Index in the CDC generation, its parent Index should also be included in the CDC generation.

- Object Type
  - Object Type is used to denote what kind of object it is.

Object Type	Comments
DEFTYPE	Denotes a static type definition such as Boolean, unsigned8, etc.,
DEFSTRUCT	Defines a record type
VAR	An object with single value such as Unsigned16, Interger32, OctetString, MAC address, etc.,
ARRAY	A multiple data field object where each field is simple variable of the same basic datatype. Eg: array of unsigned64 etc., **Note:** Subindex 0x00 is of Unsigned8 and therefore not part of ARRAY data.
RECORD	A multiple data field object where the data fields may be any combinations of simple variables. **Note:** Subindex 0x00 is of Unsigned8 and therefore not part of RECORD data.

Table 7: Object type definitions

- Data Type
  - This property provides the information about the datatype of the index/subIndex. The datatype determines the size of the value for the object. The following table provides the list of supported datatypes, its data size and whether it is allowed to be mapped to a PDO object.

Data Type	Data Size(Bits)	Allowed for PDO mapping
BIT, BOOLEAN	1*	No
INTEGER8, UNSIGNED8	8	Yes
INTEGER16, UNSIGNED16	16	Yes
INTEGER24, UNSIGNED24	24	No
INTEGER32, UNSIGNED32	32	Yes
INTEGER40, UNSIGNED40	40	No
INTEGER48, UNSIGNED48	48	Yes
INTEGER56, UNSIGNED56	56	No
INTEGER64 UNSIGNED64	64	Yes
REAL32	32	No
REAL64	64	No
MAC_ADDRESS	48*	No
IP_ADDRESS	32*	No
OCTET_STRING (32Characters max)	128*	No

Table 8: Datatype definitions

***** Refer 6.1.4 of EPSG_Specification

- PDO Mapping
  - This property of an object or subobject indicates whether an entry may be mapped to a PDO message. It can be any one of the following

PDO Mapping	Description
NO	Objects cannot be mapped to a PDO
DEFAULT*	Objects is a part of the default mapping
OPTIONAL	Objects may be mapped into a PDO
TPDO	Objects shall be mapped to a Transmit PDO
RPDO	Objects shall be mapped to a Receicve PDO

Table 9: PDO mapping list

***** Refer 6.2.1 of EPSG_Specification

- Access Type
  - This property of an object defines the access rights for a particular object.

Access Type	Description
Const	Read only access, value is const
RO	Read only access
WO	Write only access
RW	Read and write access

Table 10: Access type list

- Default Value
  - The default value is the value that is present by default in the machine for the respective object

- Value (Actual)
  - The actual value is the value that overrides the default value.

- Limit
  - This property indicates the range (high & low limits) for the value in the respective object. It depends on the datatype specified for the object or unless specified in the EPSG_Specification for the object.

- Dec/Hex radio button
  - The user can toggle between decimal or hexadecimal view of the value by choosing from the radio buttons.

4.10. Process Data Objects

- Process data objects are the objects used for isochronous data exchange between POWERLINK nodes. The Objects (0x14xx, 0x16xx and 0x18xx, 0x1Axx) are collectively known as PDO objects.
- PDO objects will be presented in a tabular structure from the user can select the values listed in the dropdown boxes.

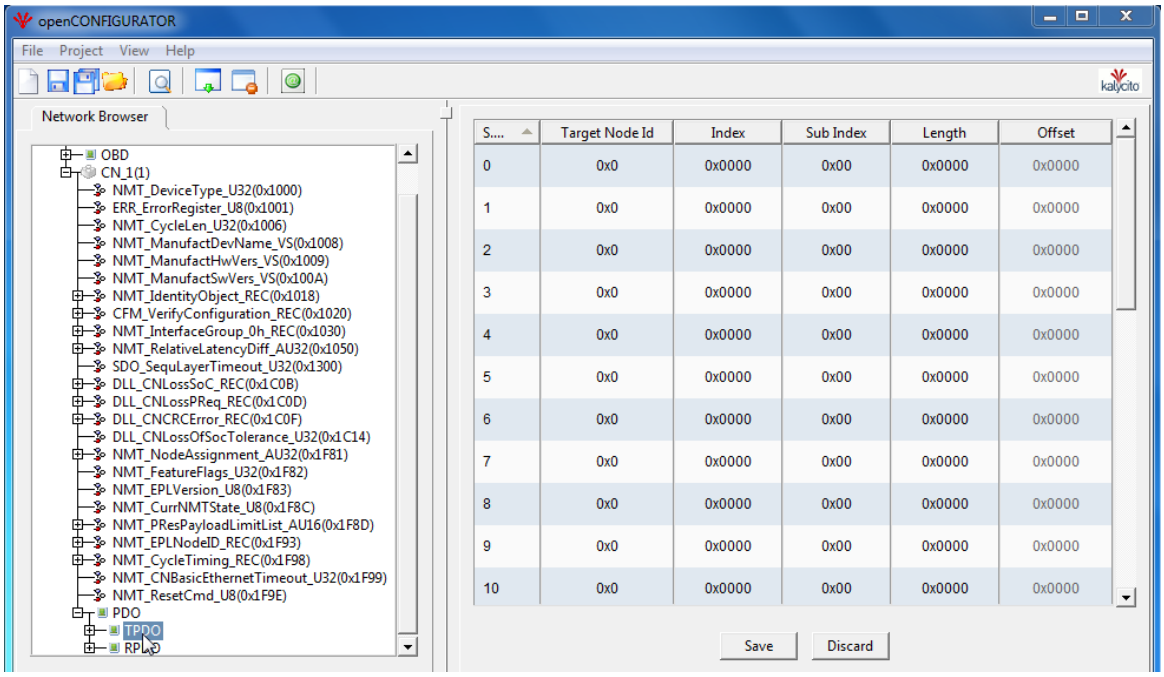

Figure 24: A sample PDO mapping table

4.10.1. Editing PDO objects

Column	Description	Allowed Range	
Target Node Id	Node Id of the PDO target	0x0	Broadcast Node Id
		0x1 – 0xEF	Available CN Node Id (Cross Traffic)
		0xF0	MN Node Id (Pres chaining)
Index	Index of the object to be mapped	0x1000 - 0x9FFF	Index/subindex that passes the mapping criteria will be listed. Refer Table 12: PdoMapping vs AccessType
Sub Index	Sub-Index of the object to be mapped	0x00*, 0x01 - 0xFE	
Length	Length of the mapped object (Bit count)	Depends on the DataType of the Index / SubIndex object	
Offset	Offset related to the start of the PDO payload (Bit count)	Cumulative sum of the payload length	

Table 11: PDO table properties

4.10.2. Pdo mapping vs AccessType

For an object to be mapped to a PDO the Object should be have the following conditions should be met.

		If an object mapped a RPDO (1600 – 16FF)				If an object mapped a TPDO (1A00)*			
		Access Type property for an object							
		Const	Ro	Wo	Rw	Const	Ro	Wo	Rw
PDO Mapping property of an Object	No								
	Default			✓	✓		✓		✓
	Optional			✓	✓		✓		✓
	TPDO						✓		✓
	RPDO			✓	✓				

Table 12: PdoMapping vs AccessType

Example:

- For an Object (0x6000) with PDOmapping=”TPDO” and AccessType=”Ro” shall be mapped only to a TPDO but not to an RPDO
- For an Object (0x6200) with PDOmapping=”Optional” and AccessType=”Rw” shall be mapped to both TPDO and RPDO

Note:
Also refer to the list of datatypes allowed for pdo mapping in Table 8: Datatype definitions
***** A CN can have only one TPDO

4.11. Delete SubIndex

SubIndex of a Index of a particular node can be deleted by clicking on the node, then right clicking on the SubIndex which has to be deleted in the expanded node tree, and clicking on the 'Delete SubIndex' option in the menu that appears as shown in the below figure

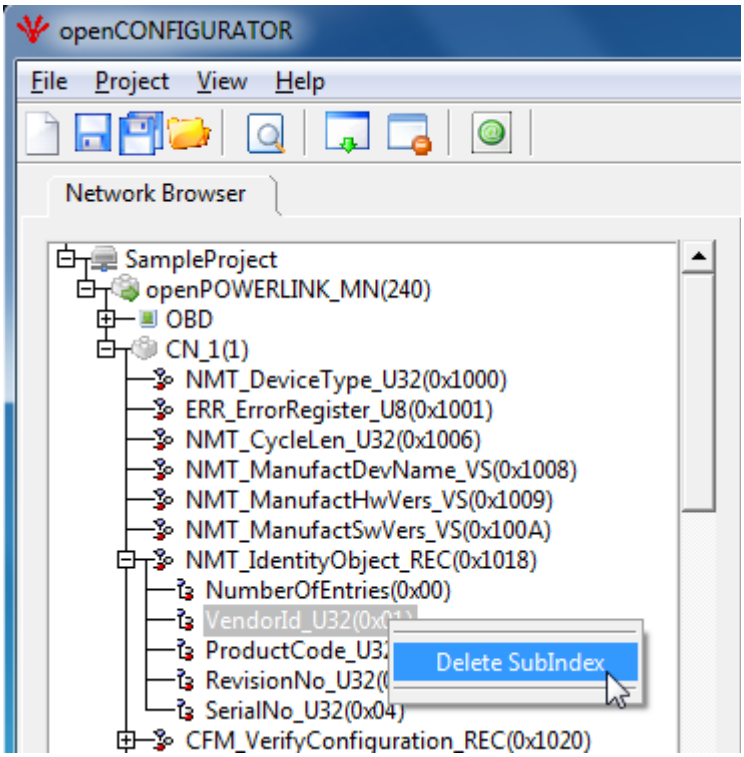

Figure 25: Delete SubIndex

Note: The SubIndex/ID 'NumberOfEntries'/0x00 cannot be deleted

4.12. Delete Index

Index of a particular node can be deleted by clicking on the node, then right clicking on the Index which has to be deleted in the expanded node tree, and clicking on the 'Delete Index' option in the menu that appears as shown in the below figure

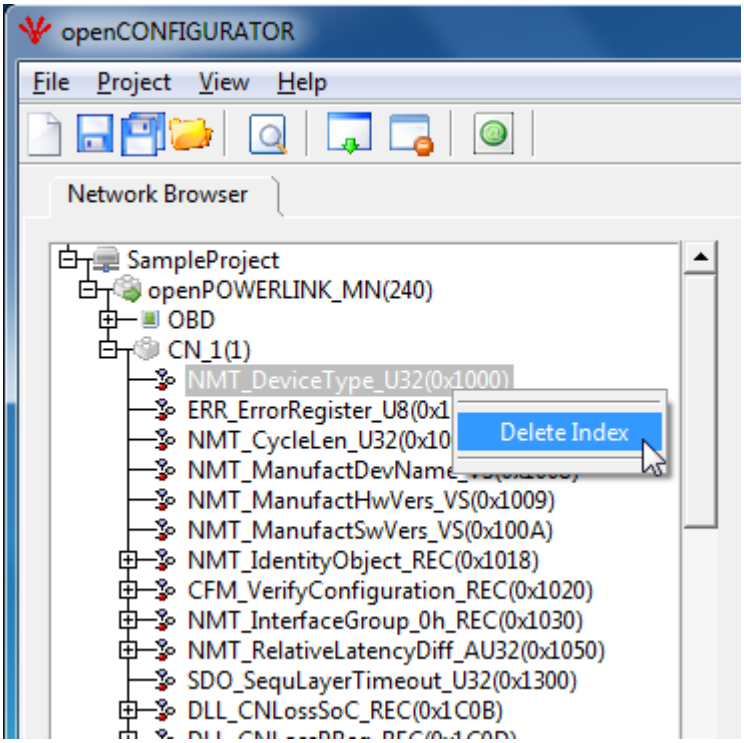

Figure 26: Delete an Index

Warning: If the user wishes to delete the index, the sub-indexes present under the index will also be deleted. The user cannot undo the operation

4.13. Delete CN Node

CN node can be deleted by right clicking on the node, a menu will appear as shown in the below figure

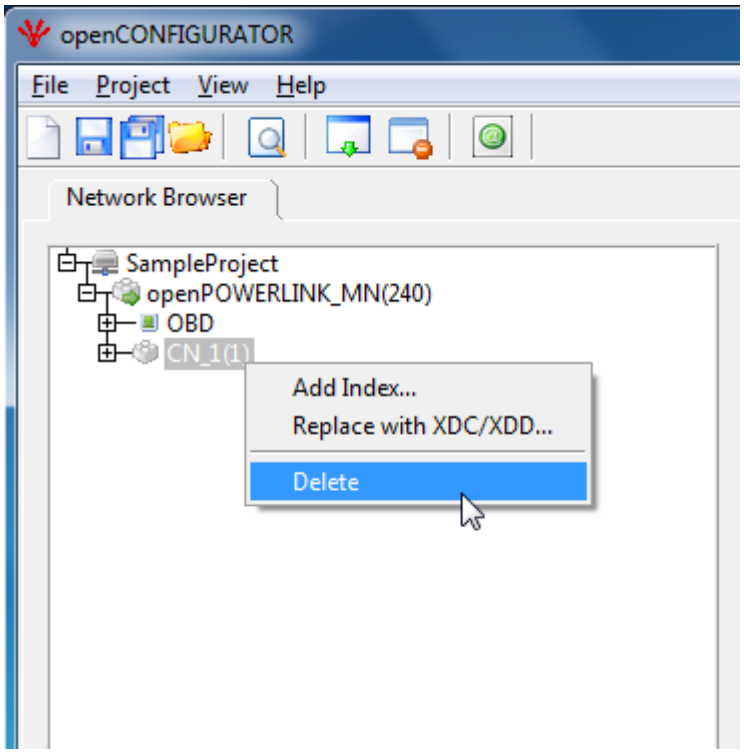

Figure 27: Delete CN Node

Warning: If the user wishes to delete the node, the Indexes and their sub-indexes will also be deleted. The user cannot undo the operation.

Tip: Instead of deleting and creating a CN, the user can replace the configuration files by choosing the **"Replace with XDC/ XDD"** submenu. This will update the node's configuration with the new configuration.

5. Output Files

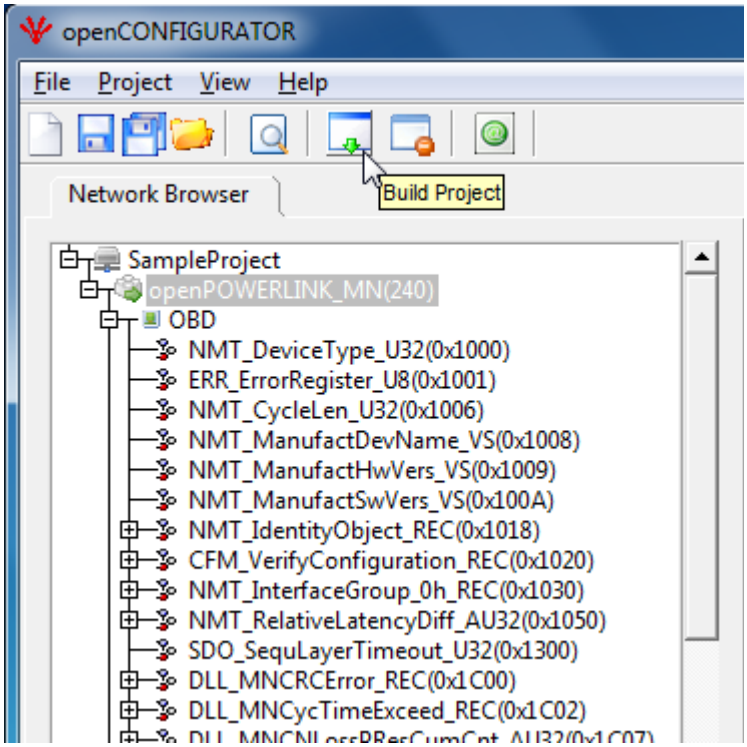

Figure 28: Build Project Icon

- Following files will be created after the successful build of project. These files will be present in the absolute path <Project location >/<Project Name>/cdc_xap folder

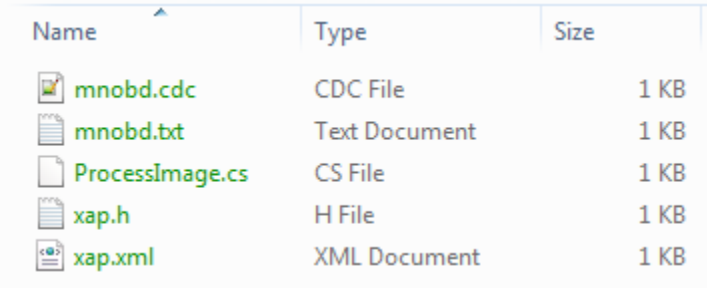

Figure 29: cdc_xap Folder View

File name	Description
mnobd.cdc	CDC binary file used with the openPOWERLINK stack
mnobd.txt	Text version of the binary CDC file
XAP.h	Header file for the application
XAP.xml	XML file with the variables names, Datatype, Datasize, ByteOffsets, BitOffsets
ProcessImage.cs	A C# namespace with the application variables and the size of the data

Table 13: Output Files

6. Uninstall

6.1.1. Linux

- Un-tar the openCONFIGURATOR_linux.tar.gz file
- Change to the directory and run the below command in the Terminal
  ./configure
- To uninstall openCONFIGURATOR, run the below command in the terminal
  sudo make uninstall

Warning: Do not run the Makefile or makefile.in by double clicking over it. This will delete all your files in the system

6.1.2. Windows (XP)

- Go to Start Menu > All Programs > openCONFIGURATOR
- Click uninstall shortcut and follow the uninstaller instructions.

Figure 30: Windows XP - Start Menu

Note: If the user had chosen not to create shortcuts during installation, you will not find the start menu entry. So you can uninstall the tool by double clicking on the Uninstall.exe from the installation directory that you've set during the installation.

6.1.3. Windows (Vista & 7)

- Right Click **'Uninstall'** and click 'Run as Administrator'

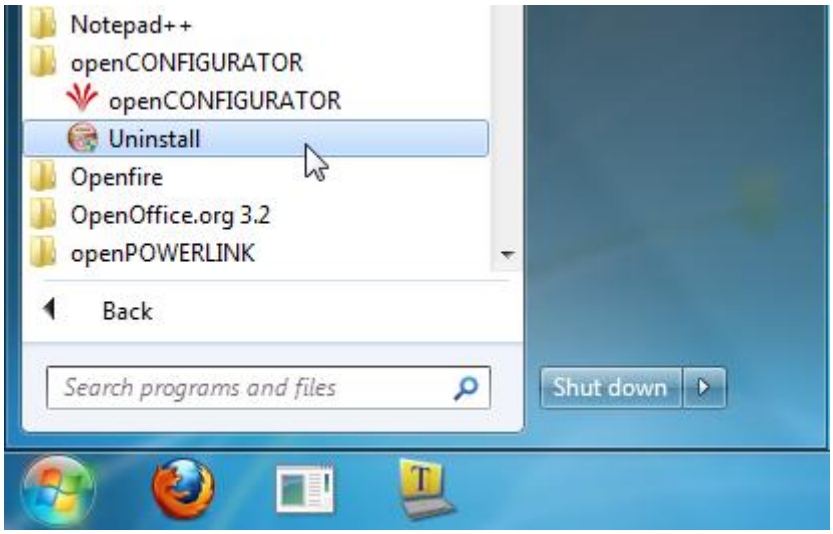

Figure 31: Uninstall - Start Menu

Note: In Vista, if Un-installation is not done as 'Administrator', the installed files will not be deleted and any further installations may not be proper. In such a case, the user shall delete the files installed in the corresponding directory.

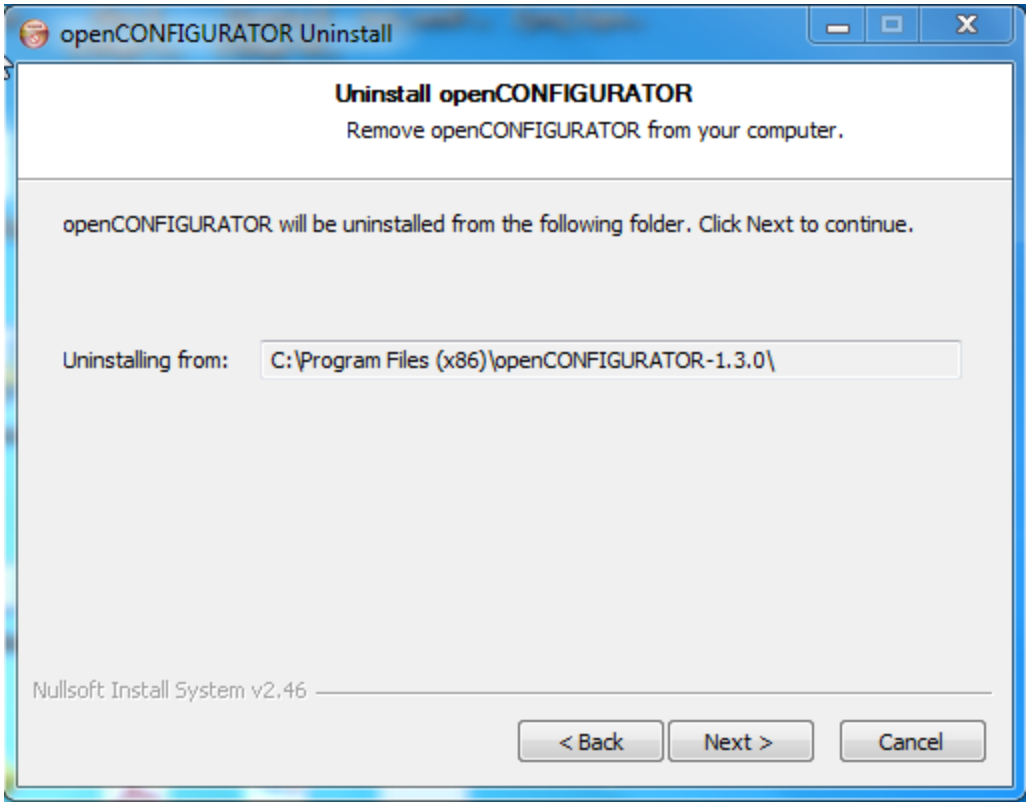

Figure 32: Uninstaller - Path

- Then press Next & then press "**Uninstall"** Button
- Press **'Close'**
- OpenCONFIGURATOR is now uninstalled

7. Compilation

For compiling the openCONFIGURATOR core & wrapper libraries follow the below instructions

7.1. Linux

7.1.1. Pre-Requisites

The user can install all the pre-requisites using the package manager in ubuntu or through terminal as "*sudo apt-get install PACKAGE*".* PACKAGE refers to any of the below package.

- Libxml2
- Libxml2-dev
- Tcl8.5
- Tk8.5
- Tclthread
- Tcl-dev
- Swig

7.1.2. Shared Library Compilation

- Unzip openCONFIGURATOR_1.3.0_src.zip
- Change to the below directory "*openCONFIGURATOR_Source_V1.3.0\openCONFIGURATORSoln\openCONFIGURATOR\src*"
- From the terminal type
    - *make all*
- The above will compile and create openCONFIGURATOR.so in the same directory
- Copy the openCONFIGURATOR.so into /usr/lib/ by using the below command.
    - *sudo cp -rvf ./openCONFIGURATOR.so /usr/lib/*
- Now change to the below directory "*openCONFIGURATOR_Source_V1.3.0\openCONFIGURATORSoln\openConfiguratorWrapper*"
- From the terminal type
    - *make all*
- The above will compile and create openConfiguratorWrapper.so in the same directory
- Copy the openConfiguratorWrapper.so into the installed directory of openCONFIGURATOR.
    - *sudo cp -rvf ./ openConfiguratorWrapper.so /usr/share/openCONFIGURATOR-1.3.0/*

Note: The compilation of wrapper needs the openCONFIGURATOR.so in /usr/lib location. If the location is changed, it should also be reflected in the Makefile for *openConfiguratorWrapper.so* An issue in any one of the above steps may lead to "Error loading shared library/ Please re-install the tool" error pop-up on the GUI.

7.2. Windows

7.2.1. Pre-requisites

- ActiveTCL 8.5
- SwigWin-2.10(It can be downloaded and installed from SWIG-Link)
- Microsoft Visual C++

7.2.2. DLL compilation

- Unzip source package and goto “openCONFIGURATORSoln.zip\openCONFIGURATORSoln”
- open “openCONFIGURATOR.sln” with MS Visual Studio Express Edition
- Install ActiveTcl mentioned in the pre-requisites and set the path of the installed directory as environmental variable
  - *TCL_PATH = “Tcl Installed Dir”*
  - Note: This may be already set by the ActiveTcl installer.
- Unzip SwigWin package and set the path to the environmental variable
  - *SWIG_PATH = “Swig installed Dir”*
- Build the solution in Release / Debug mode
- Inside the Release / Debug directory of the solution, two dlls will be created viz., openConfigurator.dll and openConfiguratorWrapper.dll
- Copy the openConfigurator.dll and openConfiguratorWrapper.dll into the openCONFIGURATOR installed path (default install path: <Program Files (x86)>/openCONFIGURATOR-V1.3.0/)

8. Txt2cdc

The users who want to edit and generate their own CDC can edit the mnobd.txt and generate the CDC with a utility (txt2cdc.exe / txt2cdc) which is available in the openCONFIGURATOR installation directory.

Steps to be followed:

- Open terminal / command prompt
- Move to the directory where mnobd.txt resides, (a sample is given below)
  - *cd openCONFIGURATOR_Projects/Project1/cdc_xap*
- Convert the edited txt file to CDC using the below command
  - */usr/share/openCONFIGURATOR-1.3.0/txt2cdc mnobd.txt mnobd.cdc*

Note: The txt2cdc executable will be found in the openCONFIGURATOR installation directory

9. FAQ's

9.1. Common Issues

- Mapping is done & mapping values are not found in mnobd.txt:
  - **Answer:** This is because of any one of the following reasons: (Consider the PDO tree node)
    - The "include in CDC" flag is not checked for each index and sub-index properties,
    - The XDD/XDC has the default value configured and not the actual value.(Only the actual value which differs from the default value will be added to the CDC)
    - Check for the value configured in the "NumberOfEntries" sub-index. This determines that number of sub-indexes to be taken into account for the PDO generation.

- Installed the new version of ActiveTCL v8.6 and the tool reports error.
  - **Answer:** Only the TCL version 8.5 is supported. ActiveTcl version 8.6 is not supported.

- Please install ActiveTCL even if the package is already installed
  - **Answer:** This is because of any one of the following reasons:
    - Check whether appropriate version of the TCL package is installed
    - Check for the Environment path that includes ActiveTcl bin path
    - Now openCONFIGURATOR will detect the TCL installation and the tool will execute. If the error is still present, follow the below steps
      - Open command prompt
      - Move to the openCONFIGURATOR installation directory using cd command
      - Type the command *tclsh openCONFIGURATOR*
- For "package thread not found" error. Please follow the below steps, this could be due to a built in TCL package installed as part of the system
  - **Answer:** Execute the command C:\Tcl\bin\tclsh.exe openCONFIGURATOR

10. References

- EPSG Draft Standard 301 v1.1.0_01 available in http://www.ethernet-powerlink.org
- XML Device Description Implementation Guidelines v1.0.0 available in http://www.ethernet-powerlink.org
- openCONFIGURATOR High level design document v1.3 available at http://www.sourceforge.net/project/openconf
- openCONFIGURATOR User quick start guide v1.3 available at http://www.sourceforge.net/project/openconf

11. Support

11.1. Release note

The ReleaseNote.txt files shipped with the openCONFIGURATOR installer package in the top level directory contain the detail release note for the current version.

11.2. Sourceforge forum

If you need help on using openCONFIGURATOR, please post on help forum at http://sourceforge.net/p/openconf/discussion/help/